#4

METHOD AND APPARATUS FOR EMBEDDING
AUXILIARY INFORMATION WITHIN ORIGINAL DATA

5

10

15

This application claims the benefit of Provisional Patent Applications Ser. #60/101,851 filed 9/25/98, #60/110,683 filed 12/02/98, #60/114,725 filed 12/31/98, and #60/126,591 filed 3/26/99, incorporated herein by reference.

This application relates to Provisional Patent Applications Ser. #60/123,587 filed 3/10/99 and #60/126,592 filed 3/26/99, incorporated herein by reference. The application also relates to Utility Patent Application Ser. #_____ filed on 9/23/99 (the same date as this application) entitled "Method and apparatus for robust embedded data" by the same author as this application, Kenneth L. Levy, incorporated herein by reference.

Field of the Invention

20

This invention relates to the field of signal processing, and more specifically, to techniques for hiding auxiliary information within original data.

Background of the Invention

25

30

35

With the recent explosive growth in the use of electronic information, enforcement of copyright laws has become more difficult. The cost of the equipment required to copy digital data representing music, art, and other valuable information has been decreasing, while the capacity of readily available data storage media has been increasing. Inexpensive devices can write enormous amounts of data to digital storage media such as writable compact disks (CD-R or CD-RWs), multi-gigabyte hard disk drives, high capacity removable magnetic disks, and soon to be available digital

versatile disks (DVDs). Readily available high-resolution printers and scanners bring the digitization and reproduction of graphic information within the means of most consumers. In addition, readily

5 available high-resolution sound cards, including analog-to-digital and digital-to-analog converters, bring the digitization and reproduction of audio information within the means of most consumers. Not only is copying digital files simple and inexpensive,

10 the Internet facilitates unauthorized distribution of copyrighted works.

Unlike analog copies, which are always inferior to the original, a copy of digital information can be identical to that of the original, with no degradation

15 due to copying. Millions of dollars are lost annually due to illegal but exact duplications of digital media and near-exact duplications of analog media. Because copying equipment is readily available, catching persons making unauthorized copies can be difficult.

20 Even if an unauthorized copier is apprehended, the creator of the original document must still prove that the allegedly unauthorized copy was in fact copied from his original work and not independently created.

One approach to solving the problem of illegal

25 copying is embedding or hiding authentication informa-tion or copy protection information within the original data. Hiding auxiliary information in original data, also called steganography, has been used for thousands of years. In steganography, a message is hidden within

30 another object or media, so that the message is minimally perceived by an observer. Steganography is related to, but different from, cryptography, in which the existence of a message is typically obvious, but its meaning is not ascertainable without special

35 knowledge.

Hidden data, also referred to as auxiliary or embedded data, can be used to prevent unauthorized copying by embedding in the original data commands that are readable by the copying device and that instruct the copying device not to make a usable copy. Hidden data can also be used to authenticate data, that is, to prove authorship. One such technique entails embedding auxiliary information in an original work in such a manner that special knowledge, such as a secret algorithm or code, is required to detect and/or remove the auxiliary information. The copier would not be able to remove the authentication information, and the original creator could prove his authorship by retrieving the embedded information, which would identify him as the author.

Data hiding has uses besides the prevention and detection of unauthorized copying. One such use is content enhancement, that is, adding information to the original data to enhance the content. For example, lyrics could be embedded in audio data on a CD. The lyrics could be viewed in a special kareoke machine, while the audio could be played on an existing CD player. Hidden data could also be used to associate different segments of video data with different viewer-selectable versions of the video on a DVD. For example, a viewer could select between a version edited for children or an unabridged version, and embedded auxiliary data would indicate to the DVD player which video segments to skip and which to include for the selected version.

The original data in which the auxiliary data is hidden may represent any type of information that is perceivable with the aid of a presenting device. For example, the data may represent music which is presented using a compact disk or audio DVD player, a video film

that is presented on a DVD player, or an image that is
presented on a computer screen or a printer.

When the combined data is presented to a user by a
normal presentation device, the auxiliary data should

5   not interfere with the use of the original data.
Ideally, the user should not be able to perceive the
auxiliary data at all. Unfortunately, increasing the
amount of the embedded auxiliary data or its
robustness, that is, its persistence to attack and data

10   transformation, may incidentally increase its
perceptibility. The degree to which the auxiliary data
can be perceived without having an adverse impact on
the user varies with the application. For example, in
CD quality audio, a minor change from the original data

15   might result in unacceptable audio artifacts. In video
data, a minor change in a presented image may be
acceptable, even though the change might be noticeable
if the original and combined works are presented and
compared side by side.

20   Several techniques are known for hiding auxiliary
information in original digital data. Data can be
hidden in original data as headers or trailers appended
to the original data. Such techniques are of limited
use in protection of copyrighted works, because the

25   auxiliary data is easily located and stripped out of
the copy, as when changing format. More sophisticated
techniques distribute the auxiliary data through the
original data, entwining the auxiliary and original
data until the auxiliary data is difficult, or even

30   statistically impossible, to identify and strip from
the combined data.

Most data hiding techniques that distribute the
auxiliary data through the original data are
computationally intense and therefore expensive to

35   implement. Many of these techniques are based upon

adding or subtracting periods of pseudo-random noise
(PN) sequences with the signal to represent the
auxiliary information, and these sequences may require
filtering (a.k.a. shaping) in the frequency domain.

5   The rest are based upon adding the auxiliary
information to the original data after the original
data has been transformed into the frequency domain,
such as by a Fourier transform. Auxiliary information
can be added in the frequency domain so that the energy

10   of the auxiliary data is spread across many frequencies
in a manner similar to that of the PN sequence. In
addition, auxiliary information can be added to the
phase of the frequency components with and without
spreading the information across frequencies.

15   Unfortunately, transforming the data into the frequency
domain and/or shaping the energy of the PN sequence so
it is less perceivable requires intense calculations.

       The ability of users to detect auxiliary data
depends not only upon the data, but also upon the

20   characteristics of the human sense organs and the
interpretation of sensory stimuli by the brain. Some
data hiding techniques transform the original data into
the frequency domain and embed auxiliary data in a
manner such that the frequency spectrum of the original

25   data reduces the perception of embedded data. This
psychophysical effect is known as masking. The
frequency distribution of the original data is used to
determine preferred frequencies at which the embedded
auxiliary data will be less perceptible, that is,

30   masked. Others use the fact that we don't perceive
phase as accurately as magnitude in the frequency
domain.

       There are some data embedding techniques that are
less computationally intense and that still distribute

35   the auxiliary data in the original data. Such

techniques include amplitude modulation, frequency band
elimination, distinct quantization, and least-
significant bit (LSB) replacement. These techniques
embed data in predetermined locations without regard to
5   the original data and are, therefore, more likely to
produce perceptual side affects in the combined data.
In addition, the LSB replacement technique is easily
disturbed by low level noise.

The ease of retrieving embedded data varies with
10 the technique used for embedding. Some data hiding and
retrieving techniques retrieve the auxiliary data by
comparing the combined data with the original data.
Others retrieve the auxiliary information using
databases of the PN sequences that were originally used
15 to hide the data. Techniques that require that a copy
of the original data or a PN database be used to
extract the auxiliary data are of limited use in
applications in which the combined data is distributed
broadly. Such techniques are useful in some
20 applications, such as data authentication, in which the
auxiliary data is retrieved rarely and only by the
copyright owner.

Thus, it would be desirable to have a data hiding
and extracting technique that is not computationally
25 intense and that maintains a desired level of
perceptibility and robustness, and in which the
embedded data that can be retrieved from the combined
data without access to the original data.

30   Summary of the Invention
Accordingly, it is an object of the present
invention to provide a method and apparatus of data
hiding and retrieval.

An advantage of the present invention is its
35 extreme efficiency relative to the prior art. Thus,

the apparatus is less expensive.

Another advantage of the invention is that it does not require the original file or other database for extracting the auxiliary data.

5   Another advantage of the invention is that it uses an adaptive algorithm based upon psychophysics. Specifically, the invention uses the original data, in unmodified form or without complex data transformations, to determine the location to insert auxiliary

10   data.

A further advantage of the invention is that, in some embodiments, the embedded data can be imperceptible when presented with existing players.

Still another advantage of the invention is that

15   usually it uses a broadband approach that distributes the auxiliary data through the original data, thereby making the auxiliary data more difficult to detect and remove.

Yet another advantage of the invention is that it

20   provides for a minimal decrease in signal-to-noise ratio (SNR) of the combined data as compared to the original data.

Still a further advantage of the invention is that it provides non-LSB capabilities, thereby providing

25   increased robustness, that is, the auxiliary data is unchanged by some transformations of the combined data.

Yet a further advantage of the invention is that it can cause no detectable change in data's statistics, thereby making authentication information impossible to

30   identify.

Still another advantage of the invention is its versatility, in that it allows the user to set parameters that vary the perceptibility, robustness, and embedding rate so that the invention can be used in

35   a broad variety of applications.

The apparatus for the present invention includes a logic processor and storage unit, such as those that come with the standard computer or on DSP boards. These devices will act as data readers, comparer and 5 data writers, such the user's desired watermark can be embedded and/or retrieved.

The process involves embedding and retrieving auxiliary information into original data to produce combined data. The invention uses a detection 10 criterion or criteria to determine where in the original data to locate and adjust data points to carry the auxiliary information. The detection criteria is used to locate positions, referred to as local masking opportunities, in the original data at which the 15 embedding of auxiliary data will produce less perception, as compared to other simplistic processes.

When embedding the auxiliary data, the data points in the original data are investigated in accordance with the detection criteria to determine the existence 20 of local masking opportunities. The detection criterion or criteria may involve, for example, comparing the data point to a predetermined value and examining the relationship of the data point to nearby points. If the detection criteria are met, one or more 25 of the nearby points, or the data point being investigated, is changed to indicate the value of an embedded bit of auxiliary data.

Thus, although the search for local masking opportunities typically progresses point by point 30 through the data, the investigation of each point may include not only the value of that point, but also values of one or more nearby points and/or one or more relationships among the points. If the investigation of a point shows the existence of a local masking 35 opportunity, data is embedded by setting the value of

one or more of the local points, that is, either the point being investigated or one or more of the nearby points.

The value to which the nearby data points are set is typically dependent upon the data point being investigated as well as on the value of the auxiliary data bit. The data point value can be set so that it has a specified relationship with the neighboring data points. The process is continued until the original data has been traversed or no additional auxiliary data remains to be embedded.

Retrieving the auxiliary data is the inverse of the embedding process. The combined data is traversed using the detection criteria to locate the local masking opportunities. As each local masking opportunity is located, the nearby data point or points that was or were set to indicate the embedded bit is or are read to extract the embedded data. The process is continued until the combined data has been traversed.

In the preferred embodiments, a data point or points are set to a value relative to the nearby data points and not to an absolute value. Both setting data points at the local masking opportunity and setting the data point to a value related to the nearby points, rather than to a value unrelated to the original data, provide masking that reduces the perceptibility of the data. The data is extracted by determining the relationships or values of the point or points near the local masking opportunity.

For the two preferred embodiments described in detail below, only points with large values are adjusted, and by a minimal amount; thus, these embodiments are based upon the masking of a weak stimulus by an intense stimulus. The process is applicable to analog and digital data. However, both

embodiments are explained in terms of digital media due
to current switch to digital media and the ease of
understanding.

Specifically, the first preferred embodiment uses
the difference between a data point after a peak and
the peak level to carry auxiliary information, as long
as the peak is above a large threshold and the original
difference between the peak and next point is not too
great. This large threshold and minimal differences
produce the desired perceptual masking. The embedding
process adjusts the point after the above-threshold
peaks to hide the auxiliary data. Correspondingly, the
retrieving process measures the difference between each
above threshold peak level and the next data point to
retrieve the auxiliary data.

The second preferred embodiment uses the change in
slope across a positive, large, steep, threshold
crossing to hide the auxiliary information, as long as
the original change in slope is not too great yet steep
enough to accept the ensuing adjustment. Again, the
large threshold produces the desired perceptual
masking. In the implementation, the embedding process
adjusts the change in slope to embed the data, whereas
the retrieving process measures the change in slope to
obtain the auxiliary data.

Usually, the preferred embedding process
implicitly spreads the energy of the auxiliary
information throughout the original data. This
broadband approach produces data that is more difficult
to remove than sub-band approaches that place the data
in an inaudible frequency range. If desired,
parameters can be chosen so that the process produces
protected data that is statistically identical to
unmarked data. Importantly, the process can be
adjusted to produce the desired tradeoffs between

perception, coding rate and robustness to attack.

Such embodiments of the invention preferably operate on the original data without requiring any complex data transformations, such as a Fourier

5 transformation. Thus, if the original data represents information in the time domain, the data can remain in the time domain as the auxiliary data is embedded and retrieved. Of course, the invention can operate on original data of all types, such as in the frequency or

10 time-frequency domain. For example, it can be applied to MPEG data, including the MPEG 1 and 2 specification, ISO 11172-3 and ISO 13818-7 respectively, herein by reference, which exists in the time-frequency domain.

Finally, the problem of bit-rate reducing

15 techniques, known as compression, removing the watermark can be bypassed by using separate, but possibly identical, watermark procedures during the compression (a.k.a. encoding) and decompression (a.k.a. decoding) process.

20 In summary, the present invention has the advantage of being extremely efficient to implement while still maintaining a desired level of robustness, i.e., resistance to tampering, including non-LSB (least significant bit) capabilities and statistical

25 invisibility. The efficiency of the present invention means that it is cheaper to implement and can be used to embed, retrieve and search more files than a computationally intense algorithm. Further objects and advantages will become apparent from a consideration of

30 the ensuing description and drawings.

Brief Description of the Drawings

FIG. 1 is a flowchart showing, in general, the steps embedding data using the invention.

35 FIG. 2 is a block diagram showing an apparatus

used to embed or retrieve data using the process of
FIG. 1.

FIG. 3 is a flowchart showing, in general, the
steps used to retrieve the data embedded using the
5   process of FIG. 1.

FIG. 4 graphically displays the operation of a
first embodiment of the invention.

FIG. 5 is a flowchart showing the steps of
embedding data in accordance with the first embodiment
10   of the invention. The dashed lines show interaction
with the auxiliary data.

FIG. 6 is a flowchart showing the steps of
retrieving data that was encoded in accordance with the
embodiment shown in FIG. 5. The dashed lines show
15   interaction with the auxiliary data.

FIG. 7 graphically displays the operation of a
second embodiment of the invention.

FIG. 8 is a flowchart showing the steps of
embedding data in accordance with a second embodiment
20   of the invention. The dashed lines show interaction
with the auxiliary data.

FIG. 9 is a flowchart showing the steps of
retrieving data that was encoded in accordance with the
embodiment shown in FIG. 8. The dashed lines show
25   interaction with the auxiliary data.

FIG. 10 demonstrates the operation of the
invention in conjunction with digital compression
techniques.

FIG. 11 A and B are two block diagrams showing an
30   embedding and retrieving apparatus used to carry out
the invention.

FIG. 12 shows an embodiment of the apparatus of
FIG. 2 for embedding data.

FIG. 13 is a block diagram showing an embodiment
35   of the apparatus of FIG. 2 for retrieving the data.

Detailed Description

The system according to a preferred embodiment of
the present invention comprises a method and apparatus
for hiding auxiliary information (or data) in original
data and for retrieving the auxiliary information.

FIG. 1 is an overview of the steps involved in
carrying out a method of the invention to embed data.
FIG. 2 shows a block diagram of an apparatus **10** used to
perform the method of FIG. 1. Apparatus **10** includes a
logic processor **14,** which can be a general purpose
microprocessor, such as an Intel Pentium or DEC Alpha,
of the type a personal computer or engineering
workstation, a digital signal processor (DSP), such the
hte Texas Instruments TMS320 line, a specialized CPU,
such as a media processor, or a custom processing
circuit. Apparatus **10** also includes a storage unit **18,**
which can include random access memory (RAM) or delays.
Because the algorithms used by the invention are not
computationally intense, they require calculations on
the order of less than one million instructions per
second and can be performed by most modern personal
computers.

The original data mentioned below might represent
sound that is recorded by sampling its amplitude
periodically, with each sample using binary numbers to
represent the magnitude of the sound at a particular
time. Alternatively, the original data could be any
series of binary data associated into groups.
Similarly, the auxiliary information is any data that
can be represented as "1"s and "0"s.

FIG. 1 shows that in step **20**, a portion of the
original data is read into storage unit **18** of FIG. 2.
Step **24** shows that the sample data is investigated
sequentially by the logic processor **14** to locate sample

points that meet predefined detection criteria. Such sample points indicate the existence of "local masking opportunities," because the detection criteria are such that a change in the value of the sample or a few
5   samples at or near that point to embed auxiliary data will usually have minimal perceivable by the listener of the sound. The amount of masking will depend upon the data type and settings chosen by the user. For example, the masking will be great for uncompressed
10   audio and less for bit-rate reduced (digitally compressed) audio such as MPEG. The same detection criteria will be applied during data retrieval to locate the hidden data.

      Each point in the original data is preferably
15   investigated to determine whether it represents a local masking opportunity. The criterion or criteria for determining local masking opportunities may entail not only the value of the point being investigated, but may also include the value of at least one nearby or
20   neighboring point, or the relationship between the nearby point and the point being investigated. The detection criteria can require, for example, that the point being investigated exceeds a certain threshold value and/or that the point be a local maximum or peak.
25   The criteria may include a requirement that a point subsequent to the point being investigated have a value that differs from the point being investigated by less that a prescribed amount, or have some other relationship to the point being investigated.
30   The sample data points can be considered as plotted on a graph, for example with time on the x-axis and the magnitude of the sample on the y-axis. Thus, the series of data points can be considered as having a slope between any points, and the value of the slope
35   can be part of the detection criteria. The criteria

may specify, for example, that a slope defined by the point being investigated and a preceding point exceed a particular value, or that the change in slope before and after the point not exceeds a particular value.

5 The criteria could include any combination of requirements. The threshold criteria can be changed to meet the needs of specific applications without departing from the concept of the invention.

In each case, no complex data transformation is
10 required to mask the auxiliary data, so comparing a point to the detection criteria is relatively quick and inexpensive. Unlike prior art methods, which need to use distant points to convert the original data into the frequency domain to determine how to mask embedded
15 data, the present invention can determines masking opportunities using only nearby or neighboring points, that is, points that are too close to use to determine useful frequency data. Nearby points including points that are next to the point being investigated or within
20 a relatively small number of points, preferably less than 50 and more preferably less than 20. The criterion can be as simple as determining whether the point exceeds a threshold.

Step **26** shows that when a point meeting the
25 detection criteria is located, the value of a specified sample point or sample points near the local masking opportunity is changed to reflect the value of the auxiliary information to be embedded. Although the changed sample may be simply set to a particular value
30 to signify the value of the embedded bit, the new value typically depends upon the value of both the auxiliary data and the neighboring point or points that were investigated to detect the local masking opportunity. For example, the point may be set so that the change in
35 value or slope signifies whether the embedded bit is a

"1" or a "0".

When a point is set to its new value, it is
important that either the change does not prevent the
original sample point from continuing to meet the
5   detection criteria, or that this local masking
opportunity is skipped and not detected in the
retrieval process. Otherwise, the embedded auxiliary
data will not be retrievable.

Alternatively, it is possible to merely embed the
10   auxiliary bit as the least significant bit, or other,
preferably low order, bit. The embedded bit is still
masked because the location of the embedded bit was
chosen to represent a local masking opportunity, such
as when the data is larger than a prescribed threshold.

15   Step **30** shows that the process is ended at step **32**
if no additional auxiliary data needs to be embedded.
Otherwise, step **34** shows that if there is additional
data in memory, the search for local masking
opportunities continues. Step **36** shows that if all
20   data in memory has not yet been searched, additional
data is read into memory. Skilled persons will
recognize that some overlap of the data in memory is
required to prevent missing local masking opportunities
that occur at the beginning or end points of the data
25   in memory.

FIG. 3 broadly shows the steps involved in
carrying out a method of the invention to detect and
retrieve the embedded auxiliary information. Because
the same processor and memory that was used to embed
30   the data can be used to retrieve the data, although not
necessary, the steps of FIG. 3 will describe extracting
data using the hardware components of FIG. 2. Step **50**
shows that a portion of the original data is read into
storage unit **18**. Step **52** shows that logic processor **14**
35   investigates each data point to determine the existence

of a local masking opportunity. If a sample point
meets the local masking opportunity criteria, step **54**
shows that the embedded "1" or "0" bit of auxiliary
data is extracted using the inverse relationship of how
5   the auxiliary data was embedded. Step **56** shows that if
additional combined data is in the memory, the logic
processor continues to investigate the remaining points
with step **52**. Step **58** shows that if all the data in
memory has been investigated, but there is
10   uninvestigated combined data in the data file,
additional data is read into memory in step **50**.
Step **60** shows that the process is ended when all the
combined data has been investigated.

Two preferred embodiments are described briefly
15   here, and in detail below, to demonstrate the flavor of
this methodology. As shown in FIG. 4, the first
embodiment uses large, positive peaks as the detection
criteria **120** and the auxiliary information is stored in
the difference **130** between the peak and the next point.
20   As shown in FIG. 7, the second embodiment uses large,
steep threshold crossings with minimal change in slope
as the detection criteria **140**, and the auxiliary
information **150** is carried in the change in slope.

The methodology is applicable to analog or digital
25   data, even though the preferred embodiments use digital
data. For example, analog data can be sampled at the
Nyquist rate to produce digital data in which
additional information is hidden. Then, the combined
digital data can be returned to the analog domain by
30   any existing method known in digital signal processing
(DSP). The analog data now contains the embedded data,
which can be decoded by using sampling. This is just
one possible method to encode analog data with the
above methodology.

35   The methodology is also applicable to audio,

speech, images, video or any other perceivable signal.
With audio and speech, the original data could
represent pressure versus time, magnitude versus
frequency, or a specific frequency magnitude versus
5   time. With images, the original data could represent
gray code versus space, separate or combined RGB or
equivalent values versus space, or magnitude versus
frequency. Video data encompasses the image data with
an added dimension of time available. For example,
10   with MPEG bit-reduced audio or images the auxiliary
data could be embedded in scaling factors or frequency
coefficients versus frequency or time or both.

Usually one of the detection criteria is a large
threshold. With 16 bit audio, a threshold greater than
15   48 dB above the minimum value is desirable. This
threshold allows the data to be changed with minimal
perception due to masking. Masking is the
psychological term defined as the increase in threshold
for steady-state stimuli. Use of the term in this
20   invention is much broader than that definition, and
describes how one set of data reduces the perception of
other data. Specifically, for uncompressed, magnitude-
time data, the sensitivity of the sensory system
decreases with increased input level, thus the small
25   adjustment of an neighboring data point is masked by
the large value of the threshold. For bit-rate
reduced, time-frequency data, such as MPEG data, the
masking is minimal and more similar to the textbook
definition since masking has been used to reduce the
30   bit rate.

Finally, this method is applicable to data where
masking is not used, but the efficiency of the process
in that it is does not require a key, such as a PN
sequence, or original data for retrieval of the
35   auxiliary information, is an advantage. In summary,

the parameters of the detection criteria will determine
the interaction between the data rate, process
complexity and perceptual quality.

5 *Embodiment 1*

The first preferred embodiment is based upon
hiding the auxiliary information in large peaks within
the original data. In this embodiment, the auxiliary
information is preferably broken into N bit words, with
10 synchronization data placed between the words for
better error recovery. The auxiliary information does
not need to include sync pulses between the words if
robustness to noise or modified files is not needed.

FIG. 4 conceptually shows that the first
15 embodiment detects a peak or local maximum and sets the
value of the subsequent point in relation to the peak
to indicate the value of the embedded bit.

FIG. 5 includes the pseudocode in the form of a
flowchart for the embedding process. The process
20 begins by searching the original data until a positive
peak that lies above a large threshold, labeled thr,
and has a relatively small decrease after the peak,
labeled dS, is found. This process is demonstrated in
boxes **200, 210** and **220**. The detection criteria are
25 checked in the most computationally efficient order,
which includes first checking to see if the point
represent a peak since peaks are the least likely
criterion.

When a desirable peak is found, the data point
30 after the peak is adjusted according to a user defined
bit depth, b, to carry the auxiliary information.
Specifically, if it is the beginning of an auxiliary
word, the synchronization code is embedded by adjusting
the point after the peak, $x[n+1]$, to be equal to the
35 peak, $x[n]$, minus half of the maximum allowable change,

dS/2, between the peak and the next point, as shown in boxes **242, 230** and **250**. An auxiliary information bit of one is encoded by adjusting the point after the peak, x[n+1], to be equal to the peak, x[n], minus half

5 the maximum change, dS/2, and plus the half the bit depth magnitude, 2^{b-1}. Correspondingly, an auxiliary information bit of zero is encoded by adjusting the point after the peak, x[n+1], to be equal to the peak, x[n], minus the sum of half the maximum change, dS/2,

10 and half the bit depth magnitude, 2^{b-1}. This embedding of zeros and ones is shown in boxes **242, 240, 260, 270** and **280**. The next two points after embedding the data should be skipped so one does not create another peak for very slow changing (i.e. flat) data, as shown in

15 box **290**.

These steps are repeated until the auxiliary information, box **242** and **240**, has been hidden in the original data or the original data is finished.

FIG. 6 displays the pseudocode in the form of a

20 flowchart for the retrieval process of the first preferred embodiment. The process begins by searching the original data is searched until a positive peak that lies above a large threshold, labeled thr, and has a relatively small decrease after the peak, labeled dS,

25 is found. This process is demonstrated in boxes **300, 310** and **320**. Again, the search first looks for a peak to improve efficiency.

When a desirable peak is found, the difference between the peak and the data point after the peak is

30 measured to retrieve the auxiliary information. Specifically, if the peak minus the point after the peak, x[n]-x[n+1], is close to half of the maximum allowable change, dS/2, a new auxiliary word is beginning, as shown in boxes **330** and **350**. If the peak

35 minus the point after the peak, x[n]-x[n+1], is

approximately equal to half the maximum change, dS/2, minus half the bit depth magnitude, 2^{b-1}, an auxiliary bit of one is found. If this difference, $x[n]-x[n+1]$, is close to the sum of half the maximum change, dS/2,

5 and half the bit depth magnitude, 2^{b-1}, an auxiliary bit of zero is retrieved. This retrieving of zeros and ones is shown in boxes **340**, **360**, **370**, **380**, and **382**. The two points immediately after retrieving the data can be skipped as shown in box **390**.

10 These steps are repeated until the auxiliary information has been retrieved in the original data or the original data is finished.

There are three user-defined parameters, including threshold, thr; bit depth, b; and maximum allowable

15 change after the slope, dS. For 16 bit audio, the threshold is usually around 48 dB above the minimal quantization, as discussed above. For data with more bits per sample, the threshold may be increased to reduce perception. The bit depth is an indication of

20 the relative change to be made to the sample point to embed the data. Thus, the smaller the bit depth, the less disturbance of the original data, making the embedded data less perceptible to the listener, but less robust, that is, more susceptible to being lost to

25 noise or attack. Minimal perception in 16 bit audio is found when bit depths are between 1 and 6 bits. However, higher bit depths can be used if one desires more robustness to noise in trade for more perceptual degradation. The maximum allowable change after the

30 peak, dS, must be at least the desired bit depth magnitude, 2^b. On the one hand, one can gain better robustness to noise at the expense of more distortion, if dS is set to twice the bit depth magnitude, 2^{b+1}. On the other hand, if one desires to keep the threshold

35 undetectable to statistical cryptoanalysis (labeled

statistically invisible), dS should be set at 2^b, and b should be small, probably below 3 bits. If dS is not 2^b, one can use the discrepancy of the average difference between large positive peaks and their next

5 points between embedded file and regular file data to determine if the file contains embedded data or not. Finally, if dS is much greater than 2^b, the auxiliary information embedding rate will be increased, because more peaks will be found suitable for data embedding.

10 Using the principles explained above, skilled persons will be able to set the user-defined parameters to values appropriate to the requirements of a particular application.

As discussed above, the large threshold usually

15 reduces the perceivable effect of adding the auxiliary information, and may even cause the auxiliary data to be non-perceivable, depending upon the data type. In addition, many data points satisfy the small difference between the peak and data point after the peak, because

20 with a slope near 0 at the peak, the data is changing the least. This small difference means that the adjustment will be small as compared to the threshold, thus reducing the chance of perceiving the embedded auxiliary data

25 The pseudocode is shown using a buffer with what appears to be look ahead capabilities (i.e. $x[n+1]$). This makes the process easier to explain and understand. However, the process is causal, as determined by replacing n+1 with k, and keeping track

30 of the last two points, $x[k-1]$ and $x[k-2]$.

Finally, one can add more criteria to define the peak. For example, the peak extends for one more point each direction where $x[n]>x[n-2]$, $x[n]>x[n+2]$, $x[n]>x[n-3]$, $x[n]>x[n+3]$, and so on, or the peak is of

35 minimal sharpness, i.e. $x[n]-x[n-1]>5$. Both of these

criteria produce better robustness to noise and less distortion since it will take more noise to move the location of the peak, although changes in the peak criteria affect the rate at which auxiliary data can be

5   embedded.

     The embedded data density and bit rate will vary with the original data and with the user-defined parameters. For example, bit rates of between 99 and 268 bits per second were achieved in CD quality audio

10   data using a bit depth of 5 and a threshold of 5,000 (74 dB). Using a bit depth of 8 and maintaining a threshold at 5,000, the average embedding rate was 1,000 bits per second. When the threshold is lowed to 2,000 at a bit depth of 8, an average embedding rate of

15   2,000 bits per second was found.

Embodiment 2

     The second preferred embodiment hides the auxiliary information in large, steep threshold

20   crossings which do not have a large change in slope. The method is more robust to noise changing the detected location. This occurs because it is less likely that noise changes the location of a threshold crossing as compared to a peak, since a threshold

25   crossing usually has a slope larger than the slope at the peak, which, by definition, has a slope near zero. Testing with audio data has shown this embodiment, as compared to the first embodiment, to produce a lower embedded data rate and is more perceivable at a lower

30   bit depth, in trade for the robustness to noise. One will probably find the optimal embodiment dependent upon the application.

     FIG. 7 conceptually shows that data is embedded by setting the slope after the threshold crossing in

35   relation to the slope at the threshold crossing.

In FIG. 8, the pseudocode for hiding the auxiliary
information using the second preferred embodiment is
presented in the form of a flow chart. The process
begins by searching the original data until a positive,
5   large, steep threshold (labeled thr) crossing with
minimal change in slope (labeled dS) is found. This
process is demonstrated in boxes **400, 410** and **420**.

When the desirable threshold crossing is found,
the data point after the threshold crossing is adjusted
10   according to a user defined bit depth (b) to carry the
auxiliary information in the change in slope. Note
that the change in slope is defined as $(x[n+1]-x[n])-$
$(x[n]-x[n-1])$, or equivalently as $x[n+1]-2*x[n]+x[n-1]$.
Specifically, if it is the beginning of an auxiliary
15   word, the synchronization code is embedded by adjusting
the point after the threshold crossing, $x[n+1]$, so that
the change in slope is zero, as shown in boxes **442, 430**
and **450**. An auxiliary bit of one is encoded by
adjusting the point after the threshold crossing,
20   $x[n+1]$, so that the change in slope is positive by an
amount equal to half the bit depth magnitude, 2^{b-1}.
Correspondingly, an auxiliary bit of zero is encoded by
adjusting the point after the threshold crossing so
that the change in slope is negative by an amount equal
25   to half the bit depth magnitude, 2^{b-1}. This embedding
of zeros and ones is shown in boxes **442, 440, 460, 470**
and **480**. The point after embedding the data can be
skipped for efficiency as shown in box **490**.

These steps are repeated until the auxiliary
30   information has been hidden in the original data or the
original data is finished.

FIG. 9 shows the pseudocode in the form of a
flowchart for the retrieval of the auxiliary
information in the second preferred embodiment. The
35   process begins by searching the original data until a

positive, large, steep threshold (labeled thr) crossing with minimal change in slope (labeled dS), is found. This process is demonstrated in boxes **500, 510** and **520.**

When a desirable threshold crossing is found, the
5 change in slope around the threshold is measured to retrieve the auxiliary information. Again, the change in slope is defined as $(x[n+1]-x[n])-(x[n]-x[n-1])$, or equivalently as $x[n+1]-2*x[n]+x[n-1]$. Specifically, if the threshold crossing has almost zero change in slope,
10 a new auxiliary word is begun, as shown in boxes **530** and **550.** If the threshold crossing has a positive change in slope approximately equal to half the bit depth magnitude, 2^{b-1}, an auxiliary bit of one is found. If the threshold crossing has a negative change in
15 slope approximately equal to half the bit depth magnitude, 2^{b-1}, an auxiliary bit of zero is retrieved. This retrieving of zeros and ones is shown in boxes **540, 560, 570, 580,** and **582.** The point after retrieving the data can be skipped for efficiency as
20 shown in box **590.**

These steps are repeated until the auxiliary information has been retrieved in the original data or the original data is finished.

As mentioned above, one does not want the
25 embedding process to eliminate the embedded location from fulfilling the detection criteria. Specifically, in this embodiment, the pre-threshold change condition, $x[n]-x[n-1]>dS+2^{b-1}$, in the detection criteria of box **420** and **520** requires that the adjustment of the next
30 data point does not bring the point back below the threshold. An alternative approach, is to ignore this condition and to set either the current or next point $(x[n]$ or $x[n+1]$, respectively) to the threshold if the embedding process would cause the next point to move
35 below the threshold, and ignore any data points that

are equal to the threshold in both the embedding and retrieving process. Interestingly, only when embedding a sync or 0 could the next point move below the threshold. Given these options, the described

5   embodiment is chosen so the process is causal, thus incorporating the known advantages of causal processes.

Once again, the large threshold and maximum allowable change in slope condition, dS, reduce the perception of embedding the auxiliary data, and

10 depending upon the data type can cause the embedding process to be completely non-perceivable. The maximum allowable change in slope condition, dS, can have any value. A larger value allows a higher data rate with more perceivable distortion, whereas a smaller value

15 produces minimal distortion with a lower data rate. Our preferred setting for dS in 16 bit audio is equal to the bit depth magnitude, 2^b. Again, bit depths below 6 bits produce minimal distortion, but higher bit depths can be used for robustness to noise and attack.

20 Using a threshold of 2,000 (i.e. 66 dB) and a bit depth of 5, data rates between 40-100 bits per second are expected, with an average of about 75 bits per second, for CD quality audio. At a bit depth of 8, the bit rate increases to an average of 100 bits per

25 second.

Modifications

The preferred embodiments have been described in detail above. However, there are many simple

30 modifications that can be made to optimize the process for each use. Thus, these modifications and many similar ones produce a process that is equivalent to the one taught in this disclosure.

In some applications, a very simple embodiment

35 could use a simple threshold to determine a local

masking opportunity and then encode the auxiliary data
in the LSB of the point exceeding the threshold or of
another point in the vicinity of the point exceeding
the threshold. Such a variation is extremely simple,
yet provides reduced perceptibility compared to prior
art LSB schemes. As with the other embodiments, one
must ensure that changing the value does not remove the
point for the detection criterion. In this case, one
could simply skip embedding where the change brings the
data below the threshold, and change the current value
of the data point to the threshold so that the data
point will be skipped in the retrieving phase.

To increase the robustness of the invention to
attack or noise, the following changes could be made.
(Attack is defined as a person or machine trying to
remove the auxiliary information from the combined
signal without distorting the perception of the
original data.)

Using a dynamic threshold can make it harder to
remove the auxiliary information. An example dynamic
threshold is an offset sinusoidal waveform. When using
a dynamic threshold, dS should be small and close to 2^b
so that the process does not change the distribution of
the differences between neighboring points, i.e. be
statistically invisible; thus, an attacker cannot use
this data to find the threshold.

One can also use the statistical gaps when dS is
larger than 2^b to find the threshold if the attack uses
a DC shift. A DC shift is obviously a more potent
attack for the second preferred embodiment than the
first, but could affect the first preferred embodiment
since threshold is one of the detection criteria.

The process could use more global definitions for
peaks and threshold crossings, for better robustness to
noise. Specifically, a peak or threshold crossing

definition that includes more points on each side.

Finally, the process can use any type of error correction in the auxiliary information to increase the robustness.

5   To increase the data rate, the following changes could be made. The auxiliary information does not need to include the extra sync pulses between the N-bit words, especially if robustness to noise is not needed. In addition, negative going peaks and/or more

10   thresholds can be used to increase bit rate. Finally, the process can use more than a binary system in adjusting the second bit to encode more information. However, the result is more likely to be perceivable or less robust to attack.

15   An interesting twist is to embed different auxiliary information on positive and negative peaks, and/or on various thresholds. In addition, with stereo files, you can code the channels separately or move between channels with consecutive points moving between

20   left and right channels.

A change that could improve the perception is to move the data point after the embedded point towards the value of the embedded point if combining the auxiliary information causes a large value change in

25   the embedded point.

As mentioned above, the data does not have to be relative to time. For example, the data could represent magnitude versus frequency. In addition, the data could be viewed as magnitude of a specific

30   frequency versus time. You could include all frequencies for an increased data rate. In other words, you could embed in the spectrum or spectrogram. Advantageously, you don't have to change the format of the given data to use this process and corresponding

35   apparatus.

For example, for bit-reduced data, such as MPEG compressed data. MPEG-compressed data comprises a series of data points that represent scaling factors and frequency coefficients. The invention could embed auxiliary data in the series of MPEG data points, using, for example, one of the two embodiments described above. When using the first preferred embodiment, one may want to increase the peak or modify its LSB such that the term is only increased, rather than decrease the point after the peak, such that quantization error is not increased in the MPEG data, especially when dealing with scaling factors. Skilled persons will recognize that, in using data like MPEG data that is divided into time frames, one could use, for example, scaling factors or frequency coefficients from consecutive frames, as well as data points representing scaling factors and coefficients of different frequencies within a frame, when determining where to embed data. For example, the coefficients for a particular frequency in consecutive frames could be considered as a series of consecutive data points, and those data points analyzed in accordance one of the embodiments above to determine where to embed data in the series. In an alternative example, the series of data points representing scaling factors or frequency coefficients for different frequencies within a frame can be analyzed in accordance with the first or second embodiment above to decide where to embed data.

The benefits of the inventions can be obtained using any series of data points, which can be analyzed to determine where to embed data, although the degree of masking may vary depending upon what the data points represent.

Example Utilizations

Below are included some example utilizations of the algorithm to aid in its understanding. This list is not complete, and only highlights the usefulness of

5 the invention. The invention in its various forms is useful in any application in which it is desirable to embed auxiliary data into original data in a minimally perceptible or imperceptible manner.

The process can be used to embed copyright

10 information. This information could include a code to determine if the data can be copied. Copying devices, such as CD writers, could include an inexpensive integrated circuit that could interpret to embedded data and prohibit copying.

15 In addition, author's or artist's name and affiliation can be embedded. In this utilization, the auxiliary information is small and would be repeated over and over with synchronization pulses between each duplication. Alternatively, the copy code could be

20 embedded using embodiment 1, and the creator's name and affiliation using embodiment 2.

The invention can also be used to send additional information. This information could be transmitted in ASCII or ANSI with 8 bit "words" (not to be included

25 with digital words being defined as 32 bits) and synchronization pulses between these words, if desired. The information could be a secret message, lyrics to the song, or a description of the artwork. For lyrics, this could be useful for kareoke machines and CD or DVD

30 players.

Digital Compression

The main problem with hiding data and digital compression (reducing bit rate not dynamic range) is

35 that the process of hiding data is incompatible with

digital bit-rate reducing techniques known as
compression (a.k.a. encoding and decoding). This
incompatibility occurs since the goal of data hiding is
to make the data minimally perceivable and the goal of
5   compression is to remove minimally perceivable parts.

To this end, FIGs. 10A and B demonstrates the
process for data hiding, if at some point the data must
be compressed. For example, this may happen while
transmitting the data.

10   In FIG. 10A, the auxiliary information is embedded
in the non-compressed data using the described invented
process or any other appropriate method, as shown in
box **600**. Then, when the data needs to be compressed,
the auxiliary information is retrieved via the

15   described invention or the other appropriate method,
and re-embedded in the compressed data with the
described invented process or the other appropriate
scheme, as shown in box **610**. The algorithm for data
hiding in the compressed and non-compressed data may be

20   the same algorithm, differing by only using different
original data.

In FIG. 10B, the auxiliary information is
retrieved from the compressed data using the described
invented process or any other appropriate algorithm,

25   the data is uncompressed, and the auxiliary information
is embedded in the uncompressed data using the
described invention or the other appropriate process,
as shown in box **620**. Finally, when needed, the
auxiliary information can be retrieved from the data

30   using the described invention or the other appropriate
algorithm, as shown in **630**. Once again, the algorithm
for data hiding in the compressed and non-compressed
data may be the same algorithm, differing by only using
different original data.

35

Apparatus

As described above, FIG 2 demonstrates that the invented process can be implemented via logic processor and storage unit **18**. FIG. 12 shows the implementation with a digital processor **1200** and digital memory **1210**. The digital processor **1200** may be defined as the equivalent of a digital signal processor (DSP), general-purpose central processing unit (CPU), or a specialized CPU, including media processors. A likely DSP chip is one of the Texas Instruments TMS320 product line. A CPU could include one of Intel's Pentium line or Motorola/IBM's PowerPC product line. The design is straightforward for someone familiar with the state of the art given the pseudocode in Figs 5 through 9.

In addition, as shown in FIG. 13, a person familiar with the state of the art could implement the process with analog and digital circuitry, either separate or in an application specific integrated circuit (ASIC). The analog and digital circuitry could include. any combination of the following devices: a digital-to-analog converter (D/A), comparators, sample-and-hold circuits, delay elements, analog-to-digital converter (A/D), and programmable logic controllers (PLC). Someone familiar with the state of the art given the previous description and pseudocode in FIGs. 5 through 9 could easily design the circuit.

FIG.s 11A and B show that the logic processor and storage unit typically comprise an embedding apparatus **700** and retrieving apparatus **770**. The embedding apparatus **700** includes the following. A data reader **710** to read original data **720** and auxiliary data **730**. A comparer **740**, that is, a circuit or device for comparing data points with known values or other data points. A data writer **750** to write the combined data **760** to a permanent or temporary storage media.

The retrieving apparatus **770** includes the following. A data reader **715** to read the combined data. The data reader **715** may be identical to the embedding data reader **710**, but it also may be

5   different. A comparer **745**, that is, a circuit or device for comparing data points with known values or other data points and, if necessary, producing the auxiliary bit or bits. Once again the comparer **745** may be identical or different that the embedding comparer

10   **740**. A data writer is not always necessary since the auxiliary information may be taken from memory or only displayed for the corresponding use.

Conclusions, Ramifications and Scope

15   As the reader can see from the description above and determined from testing the process with CD quality audio, this process and apparatus of hiding auxiliary information within original data is extremely efficient and has configurations which are non-perceivable.

20   These advantages are mainly due to the invented process finding locations to hide the auxiliary data without needing to transform the signal to the frequency domain and where masking may block or reduce the perception of the auxiliary data.

25   The foregoing descriptions of the preferred embodiments of the invention have been presented to teach those skilled in the art how to best utilize the invention. Many modifications and variations are possible in light of the above teaching. For example,

30   as discussed above, the peak criteria can be extended, the threshold may be dynamic, synchronization codes, error correcting codes, and any combination of peaks and threshold can be used jointly. To this end, the following claims define the scope and spirit of the

35   invention.